
How to Write an OSEHR Technical Report

Release 1.10

OSHER Custodial Agent

July 16, 2011

Abstract

This paper describes the basic guidelines for preparing technical reports to be contributed to the OSEHR Technical Journal. The purpose of this Journal is to foster innovation in the OSEHR community and to drive this innovations towards a mature state in which can be incorporated into the OSEHR software platform.

Contents

1	Introduction	1
2	Code Walkthrough	1
3	How to use the code	2
4	Conclusions	2

1 Introduction

Technical reports for the OSEHR Technical Journal are written by software developers and are intended for software developers. As such, they are required to include the source code of any components that is the subject of the report, and to include as well the unit tests and required data and parameters needed to run such tests.

A successful OSEHR Technical Report should be such that any reader can download the report and its accompanying materials and with minimal effort verify that the software works and produce useful results.

2 Code Walkthrough

In this hypothetical example, we do a walk through the source code of the contribution with the purpose of highlighting some of the key features in this source code piece.

First we say something smart about the first five lines:

```
1 #!/usr/bin/mumps
2 kill ^a
3 for i=1:1:9 s ^a(i)=1
4 write $data(^a(1)),!      // writes 1
5 write $order(^a("")),!    // writes 1
```

That is followed by an explanation of why the next three lines are required:

```
6 write $order(^a(1)),!      // writes 2
7 write $order(^a(9)),!      // writes empty string
8
9 set i=5
```

and conclude with a commentary of the last lines:

```
10 for j=1:1:5 set ^a(i,j)=j
11
12 write $data(^a(5)),!      // write 11
13 write $data(^a(5,1)),!    // write 1
14 write $data(^a(5,15)),!    // write 0
```

3 How to use the code

In this typical section we explain how a typical OSEHR developer can take advantage of the contribution described in this technical report.

Using the MUMPS interpreter, this routine can be executed by doing

```
mumps program1.m
```

4 Conclusions

In the final section we discuss potential improvements to the code, and what other uses it could have.